

# 行政院國家科學委員會補助專題研究計畫成果報告

## 移動代理人的行程規劃

計畫類別：a個別型計畫          整合型計畫

計畫編號：NSC89 - 2213 - E - 032 - 009

執行期間： 88 年 8 月 1 日 至 89 年 7 月 31 日

計畫主持人：張昭憲

共同主持人：

本成果報告包括以下應繳交之附件：

赴國外出差或研習心得報告一份

赴大陸地區出差或研習心得報告一份

出席國際學術會議心得報告及發表之論文各一份

國際合作研究計畫國外研究報告書一份

執行單位：淡江大學資管學系

中 華 民 國 89 年 10 月 30 日

# 行政院國家科學委員會專題研究計畫成果報告

## 移動代理人的行程規劃

### Migration Planning for Mobile Agents

計畫編號：NSC 89-2213-E-032-009

執行期限：88 年 8 月 1 日至 89 年 7 月 31 日

主持人：張昭憲 淡江大學資管系

計畫參與人員：吳勝祺、蔣嘉禾、張婉華 淡江大學資管系

#### 一、中文摘要

本計畫的目的在於發展移動代理人行程規劃(Migration Planning)所需之相關技術。在基本方法部分，我們發展一種由靜態規劃(Static Planning)與動態規劃(Dynamic Planning)所組成的二段式行程規劃流程。靜態規劃主要是根據與完成任務相關且不易變動的各項因素來進行初步規劃。而在動態規劃部份，主要是在移動代理人派出後，為考量網路的多變性所進行的路徑修正。我們已訂定規劃時相關的成本函數，並結合 TSP 及 greedy 演算法決定如何修正移動代理人的行程，期能以較佳的路徑完成使用者指定的工作。此外，在應用方面，我們也將行程規劃初步應用在分散式數位圖書館上，並取得預期的成果。

關鍵詞：移動代理人、行程規劃、軟體代理人

#### Abstract

The goal of this project is to find effective migration planning methods for mobile agents. We have developed two-steps hybrid planning methods, which is consisted of static planning and dynamic planning. In static planning, all the cost factors related the MA's tasks is pre-computed by using TSP-like algorithm to scheduling a travel route. In the dynamic planning stage, greedy computations are performed after the MA is dispatched to determine the next visiting node. Besides, we have also applied the planning concepts to query distribution for distributed digital libraries.

Keywords: Mobile Agents, Migration planning, Software Agent

#### 二、緣由與目的

近年來新興的移動代理人(Mobile Agent, 簡稱 MA)技術由於 MA 具有自主(autonomous)、遷移(migration)及常駐(residence)等特性，經由使用者的委託後，能主動遷移或常駐於各個資料來源節點，依序完成使用者委託的工作，因此對於網路壅塞、回應時間過長等問題能提供有效的解決之道。然而，除了 MA 平台研製、溝通協定與合作方式的制定後，接著有一個重要的研究課題，那就是 –

MA 的行程規劃(Migration Planning)。行程規劃目的在於：考量網路易變的環境，為 MA 規畫出最適當的遷移路徑(migration path)，讓 MA 以最少的代價(cost)來完成使用者指定的任務(task)。當利用 MA 來完成一特定工作時，如果沒有適當規劃 MA 的行程，而只是盲目地把 MA 四處傳送至各個提供服務的節點(node)，很可能會因為 MA 重複而無效率的旅行(travel)，導至網路更形擁塞，任務完成時間(round-off time)也會大幅增加。因此本計畫的目的即在於發展有效的 MA 行程規劃技術。

#### 三、結果與討論

##### (1) 結果

在 MA 的行程規劃方面，我們提出了混合式的規劃(Hybrid Scheduling)的概念。意即讓 MA 在旅行過程中，能隨時依事先所設定成本函數來調整其行程，並且能在 MA 派出後依需求修正其完成任務的策略。相關的詳細作法將在以下各節詳細介紹。

##### € 節點分類

為了發展可行的規劃方式，我們先對 MA 移動路徑上的各節點實施以下分類：

- (a) 必要節點(Essential Nodes)：在 MA 的行程中一定要經過的節點。
- (b) 非必要節點(Non-Essential Nodes)：MA 行程中的某些節點並非一定得經過，當 MA 完成部分任務後，這些節點就可以不必經過，這樣的節點統稱做非必要節點。

##### € 成本函數

釐清了前述 MA 行程的狀況分類後，接著要考慮的即為影響最佳化行程規劃的因子(factor)有哪些？又如何利用這些因子來組成成本函數(cost function)以供行程規劃時的參考依據？我們將所有可能之因子歸類成以下四種：

- (a) TI-OI(Time-Independent, Order-Independent)  $f_w$ ：例如節點之上站費用。
- (b) TI-OD(Time-Independent, Order-Dependent)  $f_o$ ：例如網路的頻寬。
- (c) TD-OI(Time-Dependent, Order-Independent)  $f_t$ ：例如節點上停留 MA 之個數。
- (d) TD-OD(Time-Dependent, Order-Dependent)  $f_{to}$ ：

例如網路傳輸速度。

根據上述四種因子，MA 由節點 i 移動到節點 j 的成本函數可用以下式子表示：

$$Cost^{i,j} = Cost_w^{i,j} + Cost_t^{i,j} + Cost_o^{i,j} + Cost_{to}^{i,j}$$

接下來，我們計算在 MA 行程中，分別以各節點為頂點之最適路徑成本：

$$Cost_{path}^j = Cost^{\Delta_1, \Delta_2} + Cost^{\Delta_2, \Delta_3} + \Lambda + Cost^{\Delta_{n-1}, \Delta_n} = \sum_{k=2}^n Cost^{\Delta_{k-1}, \Delta_k}$$

因此，最佳化行程規劃之 TSP 演算法為：

$$Cost_{TSP} = \min(Cost_{path}^1, Cost_{path}^2, Cost_{path}^3, \Lambda, Cost_{path}^n)$$

而在節點 i 時，Greedy 的計算法則為：

$$Cost_{greedy}^i = \min(Cost^{i, \Delta_1}, Cost^{i, \Delta_2}, \Lambda, Cost^{i, \Delta_n})$$

接下來我們將討論如何有效的結合上述兩種演算法，運用在 MA 之行程規劃中。

#### € 行程規劃方法

我們將介紹混合式規劃方法的步驟：此方法主要共分為以下二個步驟：(a) 初始路徑產生(Initial Path Generation) (b) 路徑微調 (Incremental Modification)。其作法分述如下：

(a) 初始路徑產生：這個步驟是在 MA 第一次傳送前進行，目的是事先根據使用者指定的成本函數計算出一條 MA 初始路徑。而最佳化採用的演算法則與一般靜態規劃相同之 TSP 相關演算法。由於必要節點規劃時，MA 行程中每一個節點均需一一走過，故規劃時不需考慮與節點順序無關之因子，故在必要節點時，成本函數定義如下：

$$Cost_o^{i,j} = Cost_o^{i,j} + Cost_{to}^{i,j}$$

而非必要節點在初始路徑規劃時，其需考慮所有因子。

(b) 路徑微調：這個步驟是在 MA 傳送後進行，如本章最初所述，單純的靜態規劃對易變的網路環境並不恰當。所以，本步驟的目的即是如何讓 MA 的行程規劃在開始傳送後能確實反應網路環境的變化。其作法為當 MA 欲離開一個節點時，它將會偵測目前所處節點的網路狀態感知模組(Network Sensing Module)，如果與初始規劃時的網路狀況不同時，則利用以下方式來微調路徑：

假設 MA 路徑節點次序為  $1 \rightarrow 2 \rightarrow 3 \rightarrow 4 \rightarrow 5 \rightarrow 6 \rightarrow 7 \rightarrow 8$ ，當 MA 到達節點編號 3 時，偵測到節點編號 4 之狀態與出發前有差異，故重新取得剩餘節點 (4,5,6,7,8) 之狀態，並以 Greedy 的方式取得最佳狀況的節點 (假設為 7)，插入原有之節點 4 之位置，作為 MA 之下一個前往之目標節點，故 MA 的路徑將微調為  $1 \rightarrow 2 \rightarrow 3 \rightarrow j \rightarrow 4 \rightarrow 5 \rightarrow 6 \rightarrow 8$ ，與原有之行程比較下，僅需比較原路徑成本 ( $Cost^{3,4}, Cost^{6,7}, Cost^{7,8}$ ) 與新路徑成本

( $Cost^{3,7}, Cost^{7,4}, Cost^{6,8}$ ) 之差異，一旦新路徑成本低於原路徑成本，則 MA 之行程即以新路徑為其後續完成任務之路徑，反之，MA 則維持原路徑，故就算 MA 以新路徑取代原有之路徑，但與原規劃路徑差異不大，此階段 greedy 演算法所參考之成本函數，由於考量新舊路徑之成本差異，故對於 TI-OI( $f_w$ ) 類因子，並不會隨著時間及節點順序而改變，可不予考慮，故其成本函數定義如下：

$$Cost^{i,j} = Cost_t^{i,j} + Cost_o^{i,j} + Cost_{to}^{i,j}$$

MA 被派遣到網路上，面臨的是一個多變的網路架構，而為了能夠讓 MA 實際反映網路即時狀況，且針對網路連結中斷及網路拓模改變所引起的錯誤，有能力自行處理，因此，一個具有網路感知能力的 MA 對於提昇執行的效率並降低其執行成本則具有實質上的助益。

#### € 模擬結果

為了驗證混合式規劃的有效性，我們分別利用亂數產生了 80 個節點，來模擬真實世界中，不同的服務提供者提供服務之收費標準、處理 MA 需求之能力及網路之傳遞速率。針對四類因子，我們分別以 CPU 時間之使用費(\$/sec,  $f_w$  因子)、網路接續費(\$/sec,  $f_o$  因子)、MA 個數(個,  $f_t$  因子)及網路傳輸速率(ms/byte,  $f_{to}$  因子)來表示。

我們從下圖可明顯的看出，當使用者以 PRICE 為需求考量來規劃 MA 的路徑時，混合式規劃可有效降 MA 的執行費用；而若是使用者以 TIME 為需求考量時，則 MA 的執行時間則相對的有較好的效率；而對於兼顧 PRICE 與 TIME 需求時，則 MA 執行的費用及時間均能有有效的降低。下表列出三種不同需求考量時，MA 執行費用及時間降低之比率。

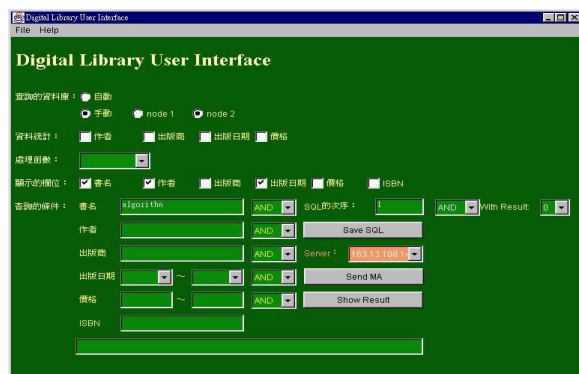
表一 MA 執行時間及費用降低之比率

	以 PRICE 為 考量		以 PRICE 為 考量		平均權重	
	時間	費用	時間	費用	時間	費用
混合式 vs.未規劃	2.39 %	45.22 %	28.47 %	23.02 %	16.40 %	44.61 %
混合式vs. 靜態規劃	1.78 %	11.71 %	23.91 %	17.12 %	11.99 %	11.53 %

#### € 應用

本計劃也已經初步將 MA 行程規劃方法應用在分散式數位圖書館效率的提昇上。在我們所建立的分散式數位圖書館實驗平台上，使用者可以進行人工或自動的行程安排，以便能快速地完成分散式查詢的任務。在實際運作時(參考下圖)，使用者可

以自行選擇所要查詢的資料庫，或是交由 DL 伺服器代為決定。在自動化 MA 行程安排中，DL 伺服器會考慮目前各個節點的回應時間，以回應時間較短之節點作為優先之考量；使用者自訂定行程時，則必須事前決定在那一個資料節點，及所要進行的查詢描述。



圖一 分散式數位圖書館使用者介面

## (2) 討論

移動代理人的應用方興未艾，然而，縱使在網路科技如此發達的今天，相關的問題並未完全解決，本計劃所提之行程規劃即為其中之一。除此之外，如安全問題與系統容量問題仍然尚未解決。與傳統的主從架構相較，由於移動代理人被賦予較大的能力來自行完成委託的任務，且被允許在各節點執行程式，如此將使各節點將更難管制駭客型移動代理人的不合法行為。至於系統容量問題，則導因於移動代理人的執行大部分均需開啟一條執行緒 (execution thread)，如果同時有數百個使用者要求以移動代理人的方式來提供服務，則系統在數百個執行緒同時運作下，很容易就限於癱瘓。

## 四、計劃成果自評

本計劃在演算法探討上已提出一套有效的混合式的行程規劃方法，並在模擬實驗中驗證其效能，相信在未來的實際應用上將可得到更進一步的驗證。目前，有關的應用已包含分散式數位圖書館及網路拍賣，並均以獲得預期的初步成果。

然而，由於網路的多變性，可能會使靜態行程規劃結果失效，而完全依賴動態規劃。如此一來，網路感知模組的偵測是否準確將會嚴重影響規劃的結果。此外，由於成本因子的計算均以量化方式進行，然而有某些因素 (如網站提供的服務優劣) 卻不容易加以量化，這將會使使用者在委託 MA 時難度增高。這些將是我們未來努力的方向。

## 五、參考文獻

- [1] V. A. Pham and A. Karmouch, "Mobile Software Agents: An Overview," IEEE Communications Magazine, July 1998, pp. 26-37.
- [2] W. Caripe, G. Cybenko, K. Moizumi and R. Gray,

"Network Awareness and Mobile Agent Systems," IEEE Communications Magazine, July 1998, pp. 44-49.

[3] J. E. White, "Mobile Agents", in Software Agent, AAAI/The MIT Press, 1997.

[4] C.-Y. Chang and J.-S. Chang, "A Visual Mobile Agent System with Itinerary Scheduling", Proceedings of the 4th International Conference on Autonomous Agents, Jun., 2000.

[5] C.-Y. Chang and J.-S. Chang, "VAMS: A Visual Mobile Agent System", Proceeding of the Fifth International Conference on Computer Science and Informatics, Atlantic City, NJ, USA, Feb 27 - Mar 3, 2000.

[6] A. Paepcke, M. Q. Wang, C.-C. Chang, S. Cousins and H. Garcia-Molina, "Using Distributed Objects to Build the Stanford Digital Library Infobus", IEEE Computer, Feb. 1999, pp 80-87.

[7] A. Paepcke, S. Cousins and H. Garcia-Molina, S. Hassan, S. Ketchpel, M. Roscheisen and T. Winograd, "Using Distributed Objects for Digital Library Interoperability", IEEE Computer, May 1996, pp. 61-68.

[8] B. Schatz and H. Chen, "Digital Libraries: Technological Advances and Social Impacts", IEEE Computer, Feb. 1999, pp. 45-50.

[9] B. Schatz and H. Chen, "Building Large-Scale Digital Libraries", IEEE Computer, May 1996.

[10] H. Zhu, T. Yang, Q. Zheng, D. Watson, O. Ibarra and T. Smith, "Adaptive Load Sharing for Clustered Digital Library Servers", Proceedings of the 7th International Symposium on High Performance Distributed Computing, 1998, pp. 235-242.

[11] N. Dushay, J. C. French, C. Lagoze, "Using Query Mediators for Distributed Searching in Federated Digital Libraries", 1999 ACM Conference on Digital Libraries, 1999, pp. 171-178.

[12] P. C. Weinstein, W. P. Brimingham and E. H. Durfee, "Agent-based Digital Libraries: Decentralization and Coordination", IEEE Communication Magazine, Jan 1999, pp100-115.